*Перечисляемый тип* (*enumerable type*) – это тип, который имеет экземплярный метод GetEnumerator(), возвращающий перечислитель. *Перечислитель* (*enumerator*) – объект, обладающий свойством Current, представляющим текущий элемент набора, и методом MoveNext() для перемещения к следующему элементу. Оператор foreach получает перечислитель, вызвав у объекта метод GetEnumerator(), а затем использует MoveNext() и Current для итерации по набору. Способы сделать класс перечисляемым:

* + Реализовать интерфейсы IEnumerable и IEnumerator.
  + Реализовать универсальные интерфейсы IEnumerable<T> и IEnumerator<T>.
  + Способ, при котором стандартные интерфейсы не применяются

public interface IEnumerable

{

IEnumerator GetEnumerator();

}

public interface IEnumerator

{

object Current { get; }

bool MoveNext();

void Reset();

}

Свойство только для чтения Current представляет текущий объект. Для обеспечения универсальности данное свойство имеет тип object. Метод MoveNext() выполняет перемещение на следующую позицию в наборе. Метод возвращает булево значение true, если дальнейшее перемещение возможно. Предполагается, что MoveNext() нужно вызвать и для получения первого элемента, то есть начальная позиция – «перед первым элементом». Метод Reset() сбрасывает позицию в начальное состояние. При записи цикла foreach объявляется переменная, тип которой совпадает с типом коллекции.

*Итератор* (*iterator*) – это блок кода, который порождает упорядоченную последовательность значений. Итератор отличает присутствие одного или нескольких операторов yield. Оператор yield return <выражение> возвращает следующее значение последовательности, а оператор yield break прекращает генерацию последовательности. Итераторы могут использоваться в качестве тела метода, если тип метода – один из интерфейсов IEnumerator, IEnumerator<T>, IEnumerable, IEnumerable<T>.

public class Shop : IEnumerable<string>

{

public IEnumerator<string> GetEnumerator()

{

foreach (var s in \_items)

{

yield return s;

}

}

}

Итераторы реализуют концепцию *отложенных вычислений*. Каждое выполнение оператора yield return ведёт к выходу из метода и возврату значения. Но состояние метода, его внутренние переменные и позиция yield return запоминаются, чтобы быть восстановленными при следующем вызове.

[**Виды коллекций**](javascript:void(0))

Многие типовые коллекции предоставляются платформой .NET Framework. Каждый тип коллекции предназначен для определенной цели.

Следующие группы классов коллекций описаны в этом разделе:

* Классы **System.Collections.Generic**
* Классы **System.Collections.Concurrent**
* Классы **System.Collections**
* Класс **Collection** в Visual Basic

Универсальную коллекцию можно создать, используя один из классов в пространстве имен [System.Collections.Generic](https://msdn.microsoft.com/ru-ru/library/system.collections.generic.aspx). Универсальная коллекция применяется в том случае, если все элементы в коллекции имеют одинаковый тип данных. Универсальная коллекция обеспечивает строгую типизацию, позволяя добавление только необходимых типов данных.

|  |  |
| --- | --- |
| Класс | Описание |
| [Dictionary<TKey, TValue>](https://msdn.microsoft.com/ru-ru/library/xfhwa508.aspx) | Предоставляет коллекцию пар ключ/значение, которые упорядочены по ключу. |
| [List<T>](https://msdn.microsoft.com/ru-ru/library/6sh2ey19.aspx) | Представляет список объектов, доступных по индексу. Предоставляет методы для поиска по списку, выполнения сортировки и изменения списка. |
| [Queue<T>](https://msdn.microsoft.com/ru-ru/library/7977ey2c.aspx) | Предоставляет коллекцию объектов, которая обслуживается по принципу "первым пришел — первым вышел" (FIFO). |
| [SortedList<TKey, TValue>](https://msdn.microsoft.com/ru-ru/library/ms132319.aspx) | Представляет коллекцию пар ключ/значение, упорядоченных по ключу на основе реализации [IComparer<T>](https://msdn.microsoft.com/ru-ru/library/8ehhxeaf.aspx). |
| [Stack<T>](https://msdn.microsoft.com/ru-ru/library/3278tedw.aspx) | Представляет коллекцию объектов, которая обслуживается по принципу "последним пришел — первым вышел" (LIFO). |

В .NET Framework 4 коллекции пространства имен [System.Collections.Concurrent](https://msdn.microsoft.com/ru-ru/library/system.collections.concurrent.aspx) предоставляют эффективные потокобезопасные операции для доступа к элементам коллекции из нескольких потоков.

Классы пространства имен [System.Collections.Concurrent](https://msdn.microsoft.com/ru-ru/library/system.collections.concurrent.aspx) следует использовать вместо соответствующих типов пространств имен [System.Collections.Generic](https://msdn.microsoft.com/ru-ru/library/system.collections.generic.aspx) и [System.Collections](https://msdn.microsoft.com/ru-ru/library/system.collections.aspx), если несколько потоков параллельно обращаются к такой коллекции. Дополнительные сведения см. в разделах [Потокобезопасные коллекции](https://msdn.microsoft.com/ru-ru/library/dd997305.aspx) и [System.Collections.Concurrent](https://msdn.microsoft.com/ru-ru/library/system.collections.concurrent.aspx).

Некоторые из классов, входящих в пространство имен [System.Collections.Concurrent](https://msdn.microsoft.com/ru-ru/library/system.collections.concurrent.aspx) — это [BlockingCollection<T>](https://msdn.microsoft.com/ru-ru/library/dd267312.aspx), [ConcurrentDictionary<TKey, TValue>](https://msdn.microsoft.com/ru-ru/library/dd287191.aspx),[ConcurrentQueue<T>](https://msdn.microsoft.com/ru-ru/library/dd267265.aspx) и [ConcurrentStack<T>](https://msdn.microsoft.com/ru-ru/library/dd267331.aspx).

Классы в пространстве имен [System.Collections](https://msdn.microsoft.com/ru-ru/library/system.collections.aspx) не хранят элементы в виде конкретно типизированных объектов, а хранят их как объекты типа **Object**.

Везде, где это возможно, следует использовать универсальные коллекции пространства имен [System.Collections.Generic](https://msdn.microsoft.com/ru-ru/library/system.collections.generic.aspx) или пространства имен[System.Collections.Concurrent](https://msdn.microsoft.com/ru-ru/library/system.collections.concurrent.aspx) вместо устаревших типов пространства имен **System.Collections**.

В следующей таблице перечислены некоторые из часто используемых классов в пространстве имен **System.Collections**:

|  |  |
| --- | --- |
| Класс | Описание |
| [ArrayList](https://msdn.microsoft.com/ru-ru/library/system.collections.arraylist.aspx) | Представляет массив объектов, размер которого динамически увеличивается по мере необходимости. |
| [Hashtable](https://msdn.microsoft.com/ru-ru/library/system.collections.hashtable.aspx) | Предоставляет коллекцию пар ключ/значение, которые упорядочены по хэш-коду ключа. |
| [Queue](https://msdn.microsoft.com/ru-ru/library/system.collections.queue.aspx) | Предоставляет коллекцию объектов, которая обслуживается по принципу "первым пришел — первым вышел" (FIFO). |
| [Stack](https://msdn.microsoft.com/ru-ru/library/system.collections.stack.aspx) | Представляет коллекцию объектов, которая обслуживается по принципу "последним пришел — первым вышел" (LIFO). |